
TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B2646 – Informační technologie

Studijní obor: 1802R007 – Informační technologie

Návrh webového portálu určeného k evidenci majetku

Design a web portal designed to record property

Bakalářská práce

Autor:

Zdeněk Pohl

Vedoucí práce:

Ing. Miroslav Holada, Ph.D.

V Liberci 14. 5. 2012



ZDE BUDE ZADÁNÍ



Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum:

Podpis:



Poděkování

Poděkování patří Ing. Miroslavu Holadovi, Ph.D., za možnost vytvořit tento projekt, za skvělé vedení práce a za důležité postřehy či rady.

Poděkování si zaslouží i spolužák Jiří Podlipný, který navrhl design celého portálu včetně podstránek a loga.

Rád bych poděkoval i Mgr. Kateřině Bisové za korekturu práce a opravu chyb.



Abstrakt

Bakalářská práce měla za úkol seznámit se s problematikou vedení evidence majetku ústavu ITE, t.j. navrhnout, zrealizovat a otestovat daný portál s reálnými daty. Portál byl naprogramován ve skriptovacím jazyce PHP ve spolupráci s databázovým systémem MySQL. Na grafické prvky portálu byla použita technologie jQuery.

Cílem práce bylo vytvořit webový portál, který umožní spravovat veškeré položky, které jsou součástí inventáře ústavu. V systému jsou implementovány nejenom základní funkce s položkami, jako např. půjčování, likvidování, editování, ale i rozšiřující funkce jako např. synchronizace v závislosti na aktuální verzi dané inventury.

V neposlední řadě portál umožňuje spravovat i veškeré fotografie daných položek, které se přes systém dají nahrávat či mazat. Nechybí ani možnost registrace a přihlášení uživatelů, tudíž je na portálu umožněno uživateli s rolí *Administrátor* spravovat veškeré uživatele včetně jejich rolí, které jsou k těmto uživatelským účtům přiděleny.

Klíčová slova:

Web, portál, php, inventura, databáze, mysql



Abstract

The bachelor's essay had a task to introduce the problem of property records in the institute ITE. My aim was to suggest, implement and test the given portal with realistic data. The portal was programmed in the script language PHP, in cooperation with the database system MySQL. For the graphic elements of the portal there was used the technology jQuery.

The aim of the work was to create a web-portal, which would enable to administer all the positions, which are part of the inventory of the institute. In the system, there are implemented not only basic functions with the positions as for example circulation, liquidation, editing, but also the further options as for example synchronization depending on the current version of the given inventory.

Last but not least the portal also enables to administrate all the photos of given positions, which can be deleted or loaded through the system. There also is not missing the possibility of registration and log in of users, that's why it is enabled to the user with the function *Administrator* to administrate all the users including their functions, which are dedicated to these user accounts.

Key words:

Web, portal, php, inventory, database, mysql



Obsah

Prohlášení.....	3
Poděkování.....	4
Abstrakt.....	5
Abstract.....	6
Obsah	7
Seznam obrázků	8
Seznam tabulek	9
1 Úvod do problematiky	10
2 Analýza systému.....	12
2.1 Specifikace evidenčního portálu	12
2.2 Třívrstvá architektura	13
2.2.1 Prezentační vrstva	13
2.2.2 Aplikační vrstva	13
2.2.3 Datová vrstva	13
2.3 Uživatelská práva	14
2.4 Synchronizace	14
3 Návrh systému	15
3.1 Návrh databáze.....	15
3.1.1 ER model	15
3.1.1.1 Entity portálu	15
3.1.1.2 ER diagram.....	18
3.1.2 Diagram datových toků - „DFD“	19
3.1.2.1 Kontextový diagram	20
3.1.2.2 DFD 0. úroveň.....	20
3.1.2.3 DFD 1. úroveň.....	21
3.1.3 Diagram případů užití („Use case diagram“).....	23
3.2 Návrh grafické podoby portálu	26
3.3 Návrh funkcí.....	26
4 Realizace portálu	28
4.1 Funkce portálu.....	28
4.1.1 Registrace.....	28
4.1.2 Přihlášení a odhlášení	29
4.1.3 Práce s položkami	30
4.1.4 Strukturované výpisy dat	33
4.1.5 Synchronizace	37



4.1.6	Uživatelé	37
4.1.7	Generování PDF souborů – mPDF	37
4.1.8	Správa fotografií	38
4.2	Grafický design	38
4.2.1	Návrh designu	38
4.2.2	Implementace	39
5	Závěr	40
6	Použitá literatura a zdroje	41
	Příloha A	42
	Příloha B	43
	Příloha C	44

Seznam obrázků

Obr. 1 - ER diagram portálu.....	19
Obr. 2 - Kontextový diagram portálu.....	20
Obr. 3 - 0. úroveň DFD	21
Obr. 4 - DFD 1. úroveň (Správa výpůjček).....	22
Obr. 5 - DFD 1. úroveň (Správa položek).....	23
Obr. 6 - Use case diagram rolí	24
Obr. 7 - Use case diagram pro hosta	24
Obr. 8 - Use case systému	25
Obr. 9 - Registrační formulář	28
Obr. 10 - Chyba způsobená špatným formátem emailu.....	29
Obr. 11 - Chybová hláška způsobená neoprávněným přístupem.....	30
Obr. 12 - Menu umožňující spravovat položky	30
Obr. 13 - Našeptávač.....	32
Obr. 14 - Vypůjčené položky	32
Obr. 15 - Vypůjčení položky.....	32
Obr. 16 - Vrácení položky.....	33
Obr. 17 - Lišta, umožňující skrývání sloupců	34
Obr. 18 - Stránkování s rozdělením po částech	34
Obr. 19 - Strukturovaný výpis logu	34
Obr. 20 - Zobrazení obrázků	35
Obr. 21 - Výpis dle zadaného filtru.....	35
Obr. 22 - Smazat filtrování	35
Obr. 23 - Položky bez zobrazeného detailu	36
Obr. 24 - Detail položky DP014531103	36



Obr. 25 - Seznam dostupných obrázků	38
Obr. 26 - Grafická podoba přihlašovacího formuláře	42
Obr. 27 - Grafická podoba celého portálu.....	43

Seznam tabulek

Tab. 1 - Ukázka kódu (validace emailu)	29
Tab. 2 - Ukázka uložení session.....	30
Tab. 3 - Práce s cookies	31
Tab. 4 - Ukázka css	39



1 Úvod do problematiky

Každá firma, společnost či instituce, která využívá při práci určité výrobky – vlastní či pouze na úrovni skladu, potřebuje uchovávat stav těchto položek, jejich počet, cenu a další parametry, které jsou důležité např. pro výpočet celkových skladových zásob či postih zaměstnance, který položku určitým způsobem znehodnotil či ztratil.

Evidenční webový portál by měl usnadňovat práci s položkami, aby si uživatelé nemuseli psát údaje o změnách a půjčování položek na lístečky, které se nakonec mohou ztratit a tudíž uživateli budou tyto ztracené informace z lístečku k ničemu, ba naopak, aby uživatelé všechno zaznamenávali do jednoho systému, který se bude starat o samotné uchovávání dat a vypisování požadovaných informací.

Portál by neměl konkurovat velkým firemním systémům, jedná se spíše o lokální evidenční systém s funkcemi specifickými pro danou problematiku – tj. pro ústav ITE.

Základní a nejdůležitější částí práce bylo nastudovat problematiku uchovávání samotných položek, protože bylo žádoucí, aby portál pracoval tak, jak by uživatelé s položkami v praxi reálně nakládali. Bylo nutné zjistit, jaké atributy se o položkách budou ukládat a v neposlední řadě jaké funkce by měl portál implementovat a jaké problémy řešit.

Základními parametry položek, které bylo třeba uchovávat, jsou:

- a) inventární číslo
- b) cena
- c) majitel
- d) typ položky
- e) datum inventarizace
- f) umístění položky

Tyto parametry musí být na portálu správně implementovány, proto je třeba dbát hlavně na samotný návrh struktury databáze, která bude data shromažďovat.

Základem portálu je tabulka s položkami, která uchovává již zmíněná data. Nedílnou součástí jsou i záznamy uživatelů, kteří mají možnost využít funkce registrace přes webové rozhraní. Citlivé údaje, jako např. hesla, jsou přenášeny a ukládány do databáze pomocí hashovací funkce, což zajistí ochranu před odposlechnutím hesla.



Strukturou podobná tabulka je log, který uživateli poskytuje možnost zjistit kompletní změny všech položek, které proběhly na portálu.

Do systému má přístup pouze registrovaný uživatel, v opačném případě je nutné se registrovat nebo využít speciálního uživatelského jména „*host*“. S tímto univerzálním uživatelským jménem může uživatel navštěvovat pouze tzv. *zobrazovací stránky*, které nepracují s databází na úrovni zápisu, ale pouze čtení u funkcí jako např. zobrazení logu, výpis historie dané položky apod.

Základem celého webu je *index.php*, který tzv. spojuje stránky v jeden celek. Celý portál se skládá z několika funkčních bloků, které se pomocí funkce *include* vkládají do této stránky. Tyto bloky se pak uživateli zobrazují jako jeden celek. Jazyk, ve kterém byl projekt vytvořen, je poslední verze značkovacího jazyka HTML, konkrétně HTML5. Tato verze je stále ve vývoji a v této době neexistuje zatím žádná bližší specifikace tohoto jazyka, či další detailnější podmínky validity.



2 Analýza systému

Dle předem daných specifikací funkcí, které portál realizuje, je nutné navrhnout strukturu databáze či zvolit programovací jazyk. Analýza je velice silný nástroj, neboť rozkládá velký problém na dílčí problémy, které se snadněji řeší a realizují.

2.1 Specifikace evidenčního portálu

- a) Uchovávání podrobných informací o položkách (inv. číslo, cena, majitel atd.)
- b) Libovolná editace konkrétní položky jejím majitelem nebo administrátorem s možností změny libovolného parametru (kromě unikátního identifikátoru dané položky)
- c) Možnost vedení logu (podrobné statistiky o dané položce a uchovávání všech změn, které byly v rámci portálu provedeny). V logu by měla být možnost skrývání nepotřebných sloupců (atributů), včetně možnosti třídit výsledky dle zadaných parametrů (datum, cena atd. vzestupně či sestupně).
- d) Implementace funkce umožňující registraci nových uživatelů, kde uživatel po registraci obdrží výchozí práva, která má možnost změnit pouze uživatel s rolí *Administrátor*.
- e) Uživatel musí mít možnost se na portál přihlásit. Po přihlášení může uživatel měnit svoje heslo k uživatelskému účtu (za předpokladu, že uživatel není přihlášen za *Hosta*).
- f) Uživatel s rolí *Administrátor* může spravovat všechny registrované členy na portálu, tj. měnit těmto uživatelům profilové údaje bez možnosti změny uživatelského jména a hesla. *Administrátor* by měl mít možnost daného uživatele vymazat.
- g) Vlastník položky nebo uživatel s rolí *Administrátor* může položky vkládat, mazat, vypůjčovat a samozřejmě vracet či vyřazovat. Vše je v případě *Administrátora* řešeno na úrovni všech položek, v opačném případě je tato možnost vztažena pouze na položky, u kterých je daný přihlášený uživatel i vlastníkem.
- h) Speciální funkce na výpis dostupných položek, tj. položek, které nejsou vypůjčeny či vyřazeny, možnost filtrovat záznamy dle předem zadaných parametrů, jako např. filtrovat záznamy dle ceny (od-do), dle typu položky (nábytek, software...) apod.
- i) Synchronizace staré a nové inventury by měla být též implementována na portálu. Funkce by měla v závislosti na kompletní inventuře z vedení univerzity určit, zda daná položka se



v lokálních datech nachází, či o této položce není v systému ani zmínka. Systém dle tohoto zjištění zajistí editaci dané položky, vymazání či přidání položky s novými parametry.

2.2 Třívrstvá architektura

Evidenční portál využívá třívrstvou architekturu, viz [4], jako většina webových projektů a portálů.

2.2.1 Prezentační vrstva

Prezentační vrstva je prakticky vše, co vidí cílový uživatel např. v prohlížeči. Jedná se o zobrazení daných informací, formulářů, ikon, obrázků a dalších grafických prvků. Tato vrstva neobsahuje žádné funkce zpracovávající data, neboť veškeré podobné požadavky postupuje o vrstvu níž.

V prezentační vrstvě pracuje např. značkový jazyk HTML (pro svoji práci jsem zvolil nejnovější verzi tohoto jazyka a to HTML5), výjimkou v této vrstvě nejsou ani kaskádové styly CSS, které dodávají celým stránkám grafický vzhled (pozadí, obrázky, barvy...). Pro portál jsem zvolil nejnovější verzi CSS3, které mají oproti starší verzi CSS 2.1 mnoho nových funkcí. Pro lepší uživatelský vzhled – našeptávač, vyskakovací políčko s datem či zjišťování správných hodnot ve formuláři bez jeho odeslání je použita javascriptová knihovna JQuery (dostupná pod licencí [7] a [8]), resp. Framework (doplňující funkce) JQuery UI, které je taktéž vyvíjena pod stejnou licencí.

2.2.2 Aplikační vrstva

Tato vrstva je meziklánekem mezi daty v datové vrstvě a jejich zobrazení ve vrstvě prezentační. Aplikační vrstva zpracovává tato data a výsledek postupuje do prezentační vrstvy, která data uživateli zobrazí.

Struktura aplikační vrstvy spočívá v použití určitého programovacího jazyka, který tyto funkce obstará.

V evidenčním portálu jsem využil skriptovací jazyk PHP ve spolupráci s databázovým systémem MySQL.

2.2.3 Datová vrstva

Datovou vrstvu tvoří databázový systém spolu s databází, která zajišťuje konzistenci dat, umožňuje tato data uchovávat nebo měnit pomocí aplikační vrstvy. Datová vrstva je hlavním paměťovým zdrojem, neboť uchovává všechny informace, položky, uživatele, prakticky všechny záznamy, se kterými portál pracuje.



2.3 Uživatelská práva

Každému uživateli, který byl registrován na portál, je přidělena uživatelská role, která umožňuje přistupovat na určité stránky a vykonávat některé funkce, které uživateli s jinou uživatelskou rolí nejsou dostupné. Nejvyšší rolí je *Administrátor*, který může na portálu neomezeně editovat položky (včetně cizích), mazat, vyřazovat, likvidovat, přidělovat uživatelské role dalším registrovaným uživatelům či jako jediná osoba synchronizovat databázi. Nižšími rolemi jsou myšleny např. role *Vedoucího ústavu*, *Členů* či *Doktorandů*, kteří mohou oproti administrátorovi editovat pouze vlastní položky, nemohou likvidovat položky, měnit údaje dalším uživatelům apod. Implementována jsou i práva *Hosta*, který nemá prakticky přístup k funkcím zapisující do databáze, ale pouze může číst data, která se vypisují na stránce jako zobrazení logu či aktivních položek.

2.4 Synchronizace

Funkce by měla řešit sjednocení záznamů mezi portálem a kompletní inventurou, která může být dostupná např. od vedení univerzity.

Portál uchovává dané položky, jejich změny, výpůjčky apod., proto se z těchto údajů zjistí všechny dostupné záznamy konkrétní položky, která se nachází i v kompletní inventuře, a porovnají se s daty tak, aby bylo zjištěno, zda se dané položce změnila v průběhu času pouze cena nebo její typ, či byla vyřazena nebo není dostupná. Na základě tohoto zjištění systém s danou položkou provede vybranou operaci (editace, přidání nové položky či vyřazení a následná likvidace). Může nastat i stav, kdy se položka nezměnila, proto funkce tuto položku ignoruje.



3 Návrh systému

Samotný návrh portálu vychází z analýzy, která byla provedena na začátku celého projektu. Návrh slouží k detailnějšímu přístupu k daným problémům, které se snaží popsat ve směrech, kterými je možno se ubírat. Samotný návrh neslouží k realizaci funkcí, spíše ke zvolení správného postupu budoucí realizace.

3.1 Návrh databáze

Jediným paměťovým médiem na portálu je databáze, která se skládá z tabulek o daných attributech. Databáze uchovává veškerá data, se kterými pracuje portál, proto je důležité správně tuto databázi navrhnout.

3.1.1 ER model

ER modely se používají pro znázornění entit (tabulek) a jejich vztahů převážně v relačních databázích. Základ relační databáze tvoří tabulka, která uchovává určitá data (záznamy). Nutná podmínka relační databáze je existence primárního klíče, který musí být v rámci dané entity unikátní (nesmí se vyskytovat hodnota NULL či duplicitní). Tento model je důležitý pro pochopení funkčnosti systému a vztahů atributů v konkrétních entitách.

3.1.1.1 Entity portálu

Action

Entita *Action* vyjadřuje možné stavy položky, kterých může položka dosáhnout za dobu, kdy je na portálu dostupná. Stavy se nazývají:

- 1) Dostupná – v případě, že je položka k dispozici na svém místě.
- 2) Vypůjčená – položka byla půjčena jinému uživateli.
- 3) Vyřazená – položka byla vyřazena uživatelem, čeká na likvidaci.
- 4) Zlikvidovaná – položku zlikvidoval administrátor, nadále se na portálu nenachází.



Borrow

Toto je poměrně důležitá entita, která zaznamenává pohyby položek mezi uživateli – půjčování a vracení položek.

Entita uchovává záznamy jako např.:

- 1) Jaká položka byla vypůjčená
- 2) Kdo položku vypůjčil (majitel nebo administrátor)
- 3) Zda je položka již vrácena, příp. kdy byla vrácena

Discarded

Entita *Discarded* uchovává informace o vyřazených položkách či jejich následné likvidaci:

- 1) Jaká položka byla vyřazena / zlikvidována
- 2) Kdy daná operace proběhla
- 3) Uživatel, který danou změnu provedl

Institute

Entita *Institute* zajišťuje uchovávání dat, která popisují pracoviště daného uživatele. Tato entita není v systému důležitá (je spíše informativní), nicméně v případě dalšího vývoje portálu, rozšiřování funkcí a počtu uživatelů i na další ústavy TUL by se tato entita mohla uplatnit v praxi k rozpoznávání uživatelů z různých ústavů.

Item

Jde o důležitou entitu, která uchovává vždy poslední změnu položek – aktuální stav položek, které jsou v systému evidovány. Tato entita je ve vztahu prakticky se všemi ostatními entitami. Mimo běžné atributy, jako inventární číslo a cena uchovává entita i např.:

- 1) Uživatele, který položku editoval či kdo je jejím vlastníkem
- 2) Aktuální stav položky
- 3) Datum poslední změny či datum inventarizace



Log

Entita *Log* slouží pro uchovávání změn položek (editace, vyřazení, přidání apod.). Oproti entitě *Item* se zde může položka vyskytnout vícekrát, ovšem každý záznam má jiný čas či datum poslední změny. Je žádoucí, aby tato entita uchovávala stejné atributy jako již zmíněná entita *Item*, proto je struktura těchto tabulek stejná.

Většina funkcí, která je na portálu implementována, využívá tuto entitu spolu s dalšími entitami jako:

- 1) type
- 2) position
- 3) users
- 4) action

Photo

Tabulka uchovává cestu k fotografii položky, proto se v této tabulce nachází pouze atribut identifikující položku a daná cesta k obrázku. Díky vazbě k entitě *Item* se atribut, který do této tabulky ukazuje, nazývá cizí klíč, který je ale zároveň primárním klíčem (zkr. PFK).

Users

Velmi důležitá entita, která ukládá data o uživateli. Základní údaje, jako např. jméno, příjmení, email či www stránky, byly doplněny o uživatelské jméno a heslo, které se používají při přihlašování nebo uživatelská práva, díky kterým systém rozpozná, zda má daný uživatel možnost pracovat např. se všemi položkami nebo je omezen pracovat pouze na svých.

Position

V této entitě se uchovávají data o možném umístění každé položky v rámci portálu. Entita obsahuje atributy jako:

- 1) název místnosti
- 2) typové označení
- 3) výměra



Rights

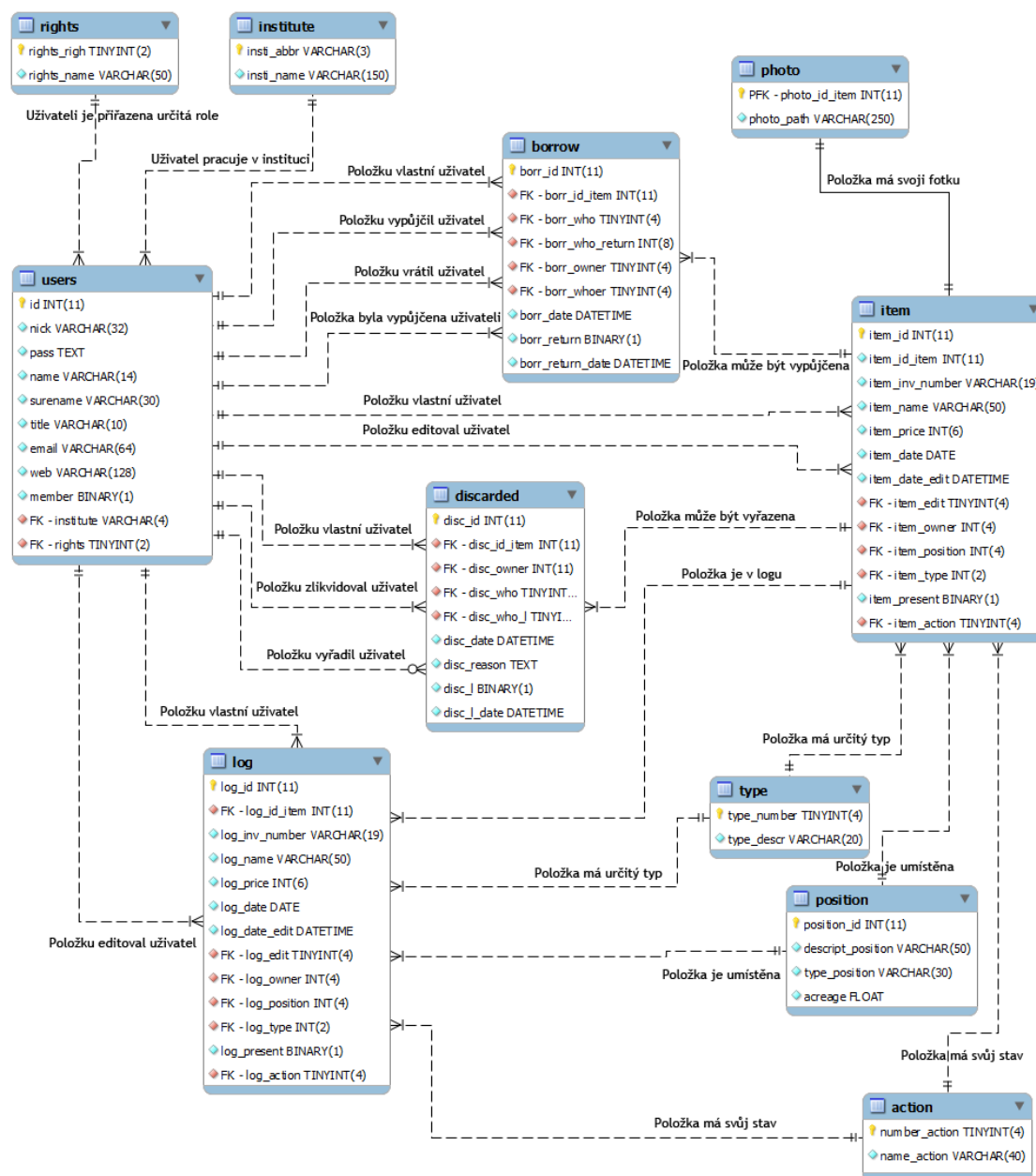
Jde o bezpečnostní entitu, která obsahuje možné uživatelské role, které lze v rámci projektu vybrat pro daného uživatele (*Administrátor, Člen, Doktorand...*). Tuto volbu nastavuje systém při registraci na nejmenší možnou hodnotu (*Host*). Možnost změnit tuto hodnotu má pouze uživatel, který má atribut určující práva nastaven na roli *Administrátor*.

Type

Entita *Type* uchovává použitelné typy předmětů, ke kterým je položka přiřazena (nábytek, elektronické přístroje, software...)

3.1.1.2 ER diagram

Výsledný ER diagram evidenčního portálu byl vytvořen v programu MySQL Workbench 5.2 včetně daných vazeb, které můžete vidět na [Obr. 1].



Obr. 1 - ER diagram portálu

3.1.2 Diagram datových toků - „DFD“

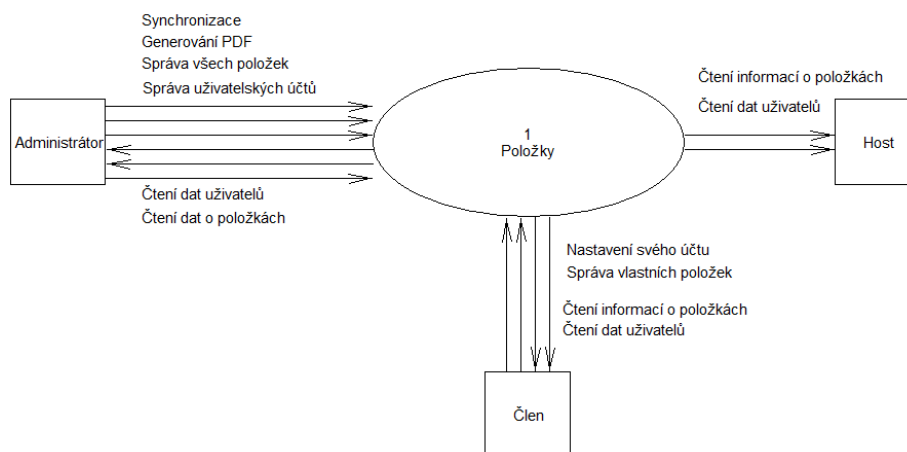
Jedná se o velice silný modelovací nástroj, který slouží k detailnímu popisu datových a řídicích toků systému. Základními stavebními prvky tohoto diagramu jsou:

- Terminátoři – uživatelé, kteří vstupují do systému
- Datové toky – vyjádření daného přesunu informací, značeny šipkou
- Datová úložiště – slouží pro uchovávání dat, např. databáze
- Funkce

Diagram datových toků se při popisu daného systému používá např. spolu s ER diagramem, který popisuje aplikaci z hlediska vlastního spojení databázových tabulek (viz 3.1.1).

3.1.2.1 Kontextový diagram

Z obrázku [Obr. 2] je patrné, že kontextový diagram popisuje obecným způsobem celý systém jako celek a zobrazuje pouze terminátory (uživatelé), kteří se systémem pracují a datové toky, které realizují dané funkce v systému.

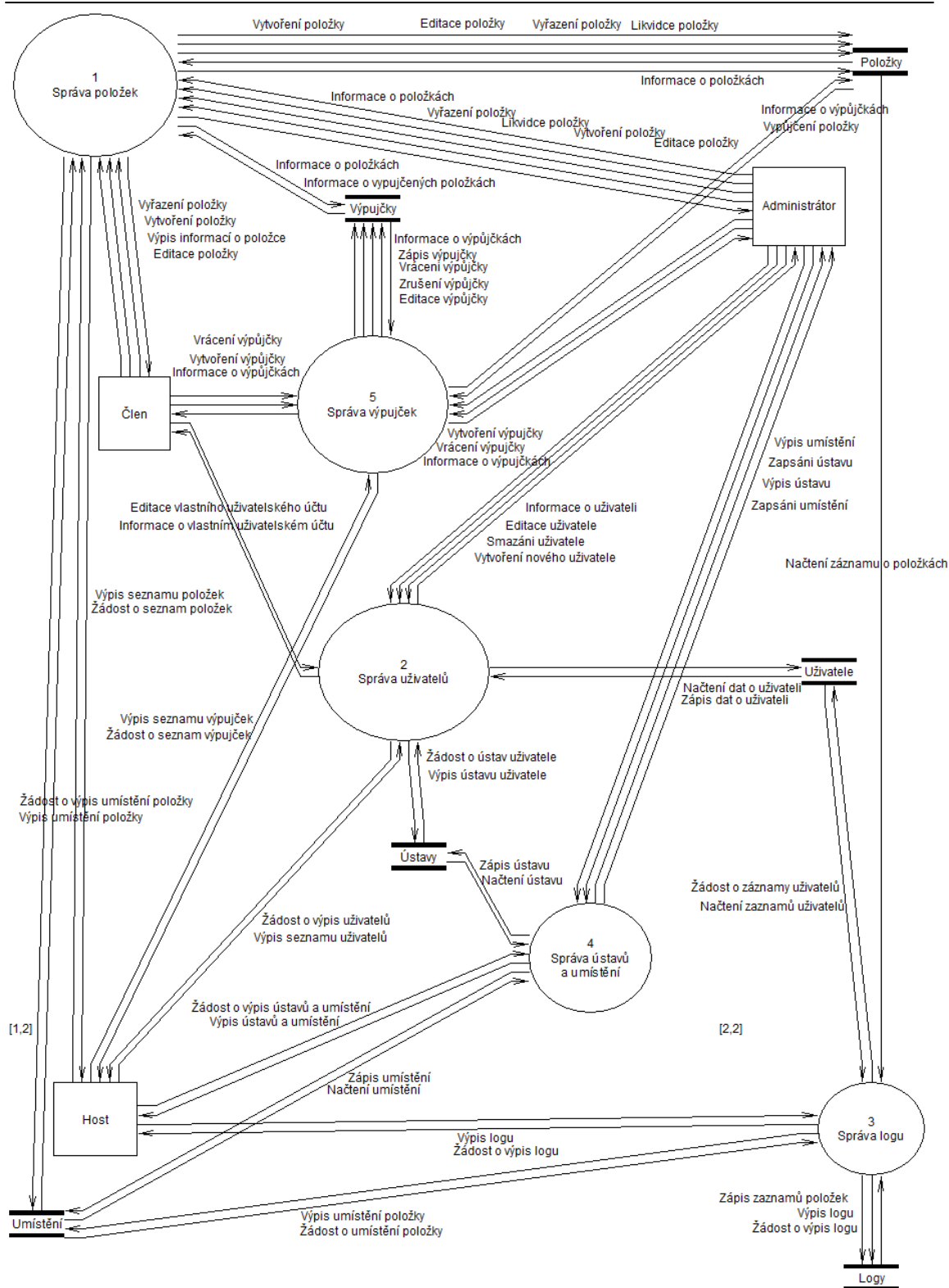


Obr. 2 - Kontextový diagram portálu

3.1.2.2 DFD 0. úroveň

0. úroveň popisuje funkce systému, se kterými pracují uživatelé. Samotné detaily funkcí, nejsou popsány v této úrovni, ale v úrovni nižší.

Na [Obr. 3] můžete vidět kompletní 0. úroveň portálu, z počtu funkcí a vazeb s terminátory je zřejmé, že každý uživatel disponuje jinými funkcemi.

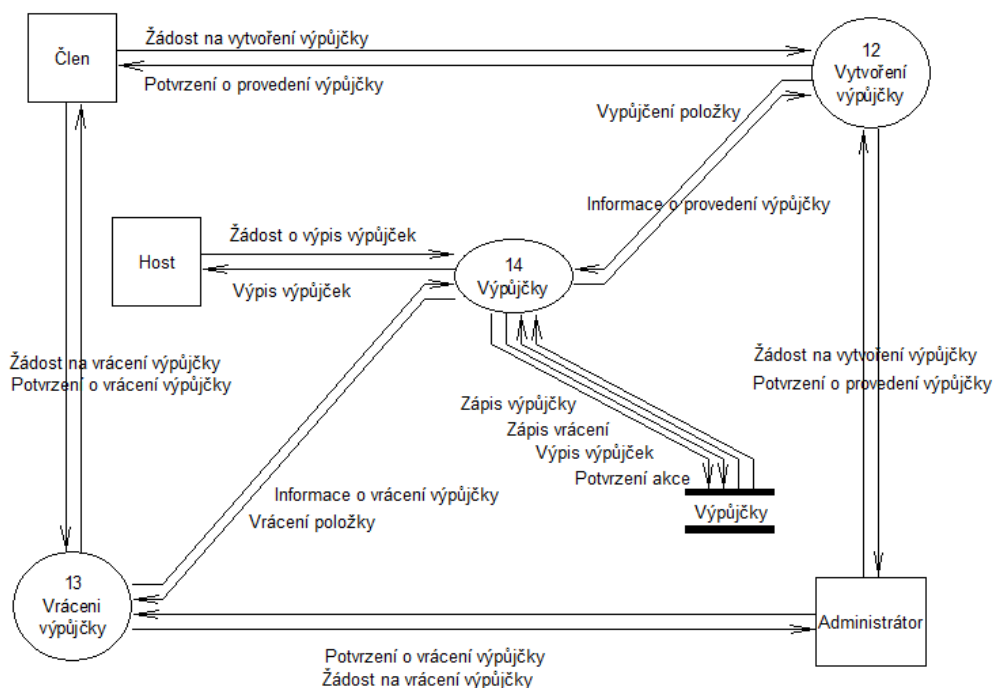


Obr. 3 - 0. úroveň DFD

3.1.2.3 DFD 1. úroveň

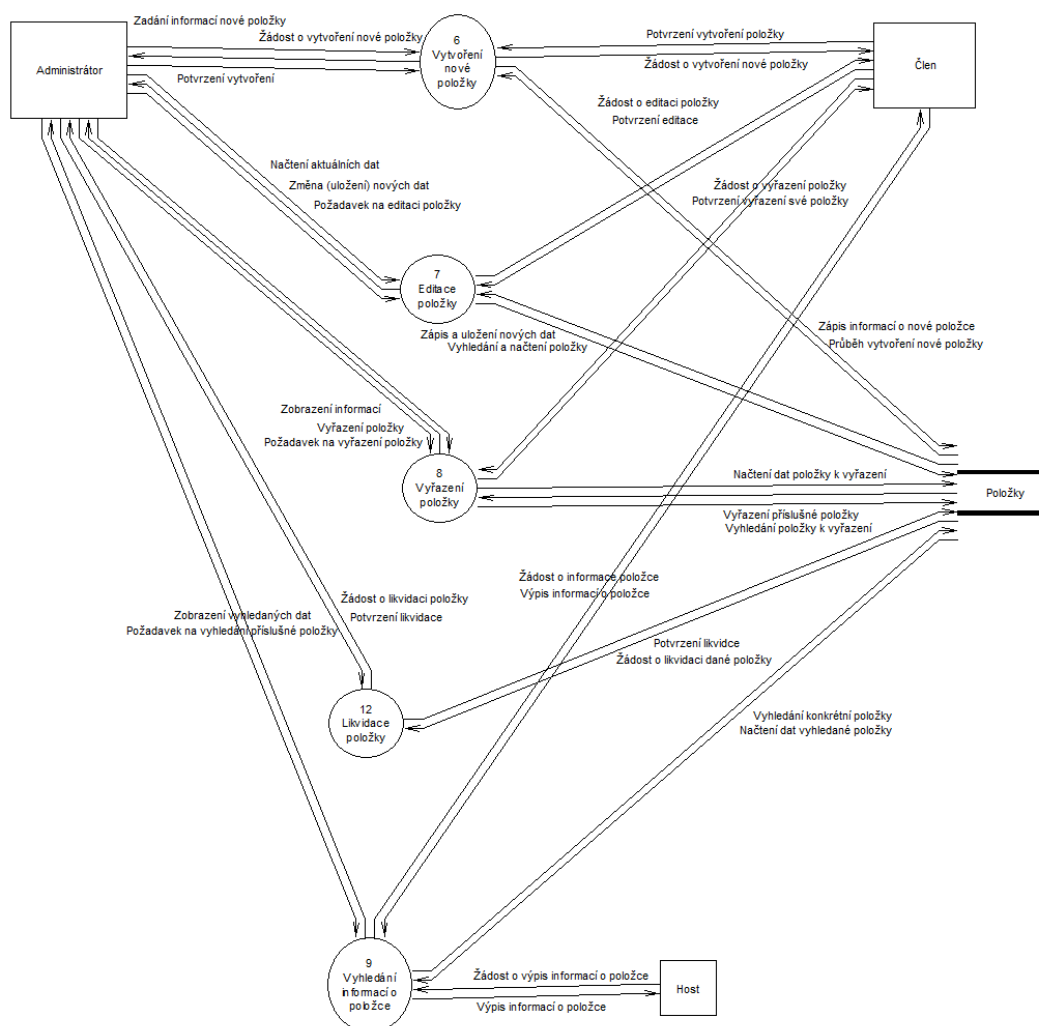
Z [Obr. 4] je patrné, že tento diagram popisuje funkci zajišťující správu výpůjček, tj. vypůjčení a vrácení položek. Uživatel, který disponuje právy, která umožňují položku vypůjčit, tj. má

uživatelskou roli vyšší než *Host*, zažádá systém o vypůjčení dané položky. Systém uloží záznam o položce, která byla vypůjčena a uživateli vrátí informativní výpis o stavu (zda došlo k chybě, nebo vypůjčení proběhlo úspěšně). Uživatel s rolí *Host*, jak je vidět na obrázku, smí pouze nahlížet do vypůjčených položek, nikoliv je sám vypůjčovat a vracet.



Obr. 4 - DFD 1. úroveň (Správa výpůjček)

Na [Obr. 5] je popsána nejdůležitější funkce, která poskytuje uživateli možnost pracovat na portálu s položkami. Základem je databázová tabulka s těmito položkami, ze které se získávají data a též se pomocí některých funkcí zapisují. Funkce umožňuje uživateli položky přidávat, editovat, vyřazovat či likvidovat včetně implementace funkce realizující pouhé získání dat o dané položce bez jakýchkoliv změn této položky. Tento obrázek popisuje i vazby mezi funkcemi a uživateli, kteří se systémem pracují. O obrázku je patrné, že *Administrátor* pracuje se všemi funkcemi, uživatel s rolí *Člen* nemá k dispozici funkci likvidace a nejnižšími právy je uživatel *Host*, který může využít pouze zobrazení informací o položkách, ale nemá možnost jakkoliv položku editovat, přidat nebo smazat.



Obr. 5 - DFD 1. úroveň (Správa položek)

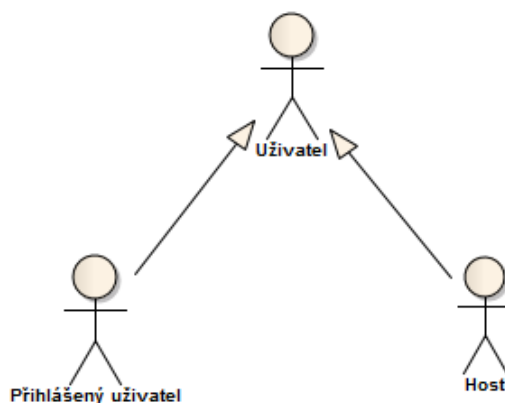
3.1.3 Diagram případů užití („Use case diagram“)

Diagram popisuje způsob komunikace systému s okolím (např. uživatelem). Zobrazuje samotné funkce systému, ale samotná implementace těchto funkcí je v tomto diagramu uživateli skryta.

Use case diagram (dále jen UC) by měl korespondovat s částí DSP (Dokument specifikace požadavků), který by měl být sepsán na základě funkcí a parametrů zadaných od zákazníka.

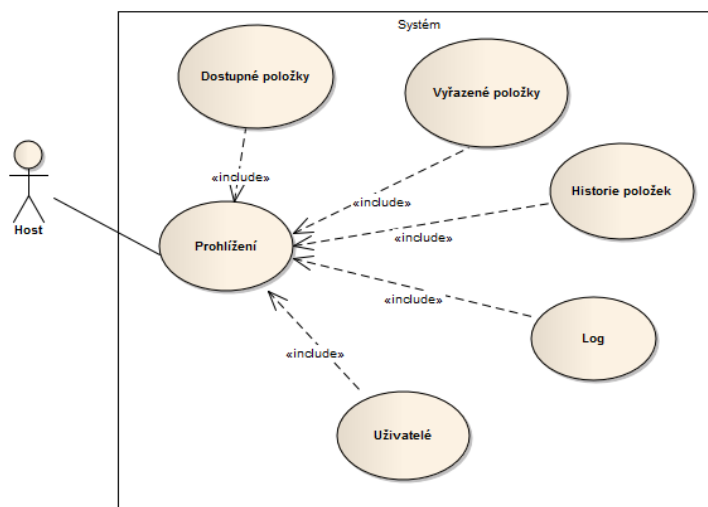
Diagram specifikuje, s jakými informacemi pracuje systém či jaké funkce používá daný uživatel.

Základní rozdělení uživatelských rolí popisuje UC, který umožňuje vystihnout patřičné vazby mezi těmito rolemi. Z obrázku [Obr. 6] je patrné, že základem portálu je Uživatel, od kterého je specifikován přihlášený uživatel a uživatel s konkrétní rolí *Host*.



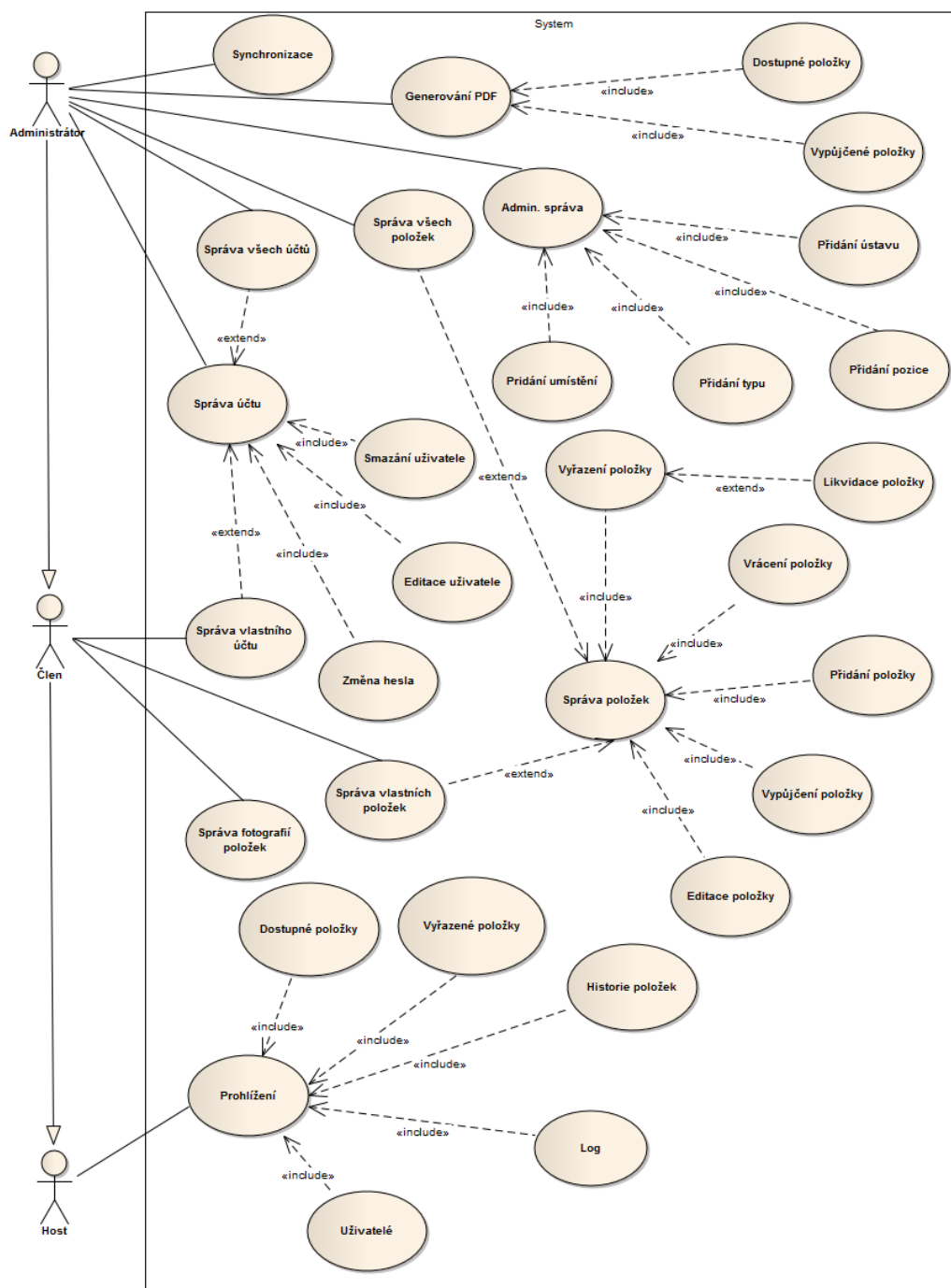
Obr. 6 - Use case diagram rolí

Uživatelskou roli *Host* vystihuje UC, který popisuje dané funkce tohoto uživatele – práva lze získat např. registrací na portálu. Z obrázku [Obr. 7] je patrné, že tato role je nejnižší na portálu, proto lze soudit, že uživatel s tímto oprávněním nebude moci na portálu pracovat s mnoha funkcemi.



Obr. 7 - Use case diagram pro hosta

Další rolí, která je důležitá pro chod portálu, je role *Člen*. Uživatel této role přejímá funkce od hosta, které jsou doplněny o možnost editace vlastního profilu či správy vlastních položek. Jak je zřejmé na [Obr. 8], *Administrátor* je nejdůležitější osoba, která má k dispozici nejvíce funkcí na portálu, neboť přejímá dostupné funkce od *Člena*, které jsou navíc rozšířeny o likvidaci položky, kompletní správu uživatelů, možnost generovat PDF, přidávání typů položek a jejich umístění či synchronizace databází.



Obr. 8 - Use case systému



3.2 Návrh grafické podoby portálu

Design je velice silný nástroj, který uživateli umožňuje lepší orientaci na stránce.

Grafická podoba webu je velice individuální, protože většinou závisí na konkrétním projektu, který realizuje dané funkce, je určen pro dané uživatele apod., ale všeobecně lze říci, že by design neměl působit na uživatele rušivě, neměl by určitě obsahovat žádné dominantní prvky, které by zbytečně upoutávaly pozornost uživatele. Téměř každý surfuje denně na internetu a každá stránka, kterou uživatel navštívil, je specifická nejenom obsahem – informacemi, kvůli kterým jsou stránky navštěvovány, ale i designem – použitím barev, symbolů, ikon a dalších grafických prvků, které se v dnešní používají.

Pravděpodobně nejrozšířenějším postupem tvorby grafické podoby webu je návrh grafiky včetně ikon, barev nebo pozadí v daném grafickém editoru, který umožňuje práci ve vrstvách.

Vrstvy jsou důležité, neboť každý prvek se dá libovolně vyjmout či uložit, proto je následné kódování do kaskádových stylů jednodušší a rychlejší, než vše řešit vyřezáváním určitých tvarů z kompletního obrázku (např. formát *jpeg*).

Grafiku lze řešit i při samotném programování, kdy se píše zároveň kód např. v jazyce HTML a zároveň se dokument styluje, což je náročnější na grafickou představu programátora, neboť výsledný design nemá na obrázku, ale při programování ho vytváří. Evidenční portál byl vytvořen nejprve jako kompletní návrh, až poté se prováděla samotná implementace grafických prvků do kaskádových stylů.

3.3 Návrh funkcí

S využitím znalosti problematiky daného portálu byly navrženy základní operace, které umožňují kompletně spravovat položky i uživatele.

Na portálu jsou implementovány základní funkce jako je registrace, kdy se uživatel registruje na portál. Samozřejmostí je i funkce, která umožňuje daného uživatele editovat či smazat (ukončení pracovního poměru atd.)

Uživatel smí spravovat pouze vlastní zaevidované položky, může editovat vypůjčovat a vracet nebo vyřazovat. Dalším stupněm vyřazení je likvidace, o kterou se stará pouze uživatel, který je na portálu *Administrátorem*.



Všem uživatelům se po přihlášení zviditelní menu s možností prohlížení logu – jak se s danou položkou manipulovalo, kdo s ní manipuloval a kdy, implementována je i funkce, která filtruje dané položky dle parametrů – majitel, cena, místnost, typ apod.

Samozřejmostí je i strukturovaný výpis vypůjčených či vyřazených položek uspořádaných do tabulky včetně výpisu uživatelů, kteří tyto změny realizovali.

Zajímavá je implementace funkce „Synchronizace“, která realizuje proces, kdy jsou do systému vloženy všechny položky, které jsou dostupné na ústavu. Funkce zjistí, která položka je oproti stávajícímu portálu dostupná, zda byla někdy vypůjčena, je nová apod. a v závislosti na těchto aspektech provede s danou položkou na portálu určitou akci – vyřadí ji, přidá ji jako novou, či změni její cenu, popř. ji ignoruje.

Doplňujícími funkcemi jsou např. výpisy položek, které jsou nejvíce vypůjčovány, výpis tří uživatelů, kteří mají zaevidováno nejvíce položek, kolik položek je kterého typu, celková cena všech položek apod.



4 Realizace portálu

Samotná realizace portálu vychází z návrhu databáze, kdy byla využita databáze Mysql, byly vytvořeny tabulky dle ER modelu a v závislosti na DFD diagram byly navrženy funkce a datové toky, které s funkcemi komunikují.

4.1 Funkce portálu

Grafická podoba webu bohužel nestačí na správnou funkčnost portálu. Je nutné implementovat funkce, které budou pracovat s danými položkami, budou umožňovat spravovat uživatele či zajišťovat komunikaci s databází.

4.1.1 Registrace

Tato funkce je důležitá na většině webových portálů. Je důležité, aby bylo důsledně odděleno, který návštěvník portálu se na něj dostal jako obyčejný uživatel či je to např. *Administrátor* nebo člověk, který se systémem pracuje.

Evidence položek neumožňuje práci s položkami, výpisem logu a dalšími funkcemi bez přihlášení. Aby se mohl uživatel na stránce přihlásit, je nejprve nutné se registrovat.

Registraci zajišťuje formulář [Obr. 9], který za použití JQuery knihovny ověřuje správný formát zadaných dat ve vstupních polích bez nutnosti formulář odesílat a tím namáhat server.

Uživatelské jméno

Heslo

Heslo znovu

Email

Web

Jméno

Příjmení

Registrovat

Obr. 9 - Registrační formulář

Ověření se provádí pomocí regulárních výrazů, které jsou umístěny v konkrétním souboru, který se na hlavní stránku vkládá jako externí skript. V případě vložení špatného formátu daného prvku je uživatel informován o chybě a důvodu, proč vznikla, viz [Obr. 10].

Obr. 10 - Chyba způsobená špatným formátem emailu

Veškeré ověření pomocí JQuery je ještě ověřeno pomocí jazyka PHP pro případ, že by si uživatel vypnul v prohlížeči javascript. Ověření správného formátu emailu zobrazuje [Tab. 1].

Tab. 1 - Ukázka kódu (validace emailu)

```
jQuery("#ValidEmail").validate({  
    expression: "  
        if (VAL.match(/^[^\W] [a-zA-Z0-9\\_\\-\\.]+([a-zA-Z0-9\\_\\-\\.]+)*\\@[a-zA-Z0-9\\_]+(\\.[a-zA-Z0-9\\_]+)*\\.[a-zA-Z]{2,4}$/) )  
            return true;  
        else  
            return false; ",  
    message: "Email má špatný formát, vložte prosím správný formát"  
});
```

4.1.2 Přihlášení a odhlášení

Uživatel se na stránce může přihlásit pomocí svého uživatelského jména a hesla, které si zvolil při registraci.

System se dotáže pomocí SQL dotazu databáze, zda existuje uživatel, který má dané uživatelské jméno a současně heslo, které bylo vyplněno. Heslo je přenášeno v zahashovaném stavu, což zaručuje, že nebude volně čitelné. Vzhledem k tomu, že je protokol HTTP bezstavový, systém musí určitým způsobem uchovávat informace, jaký uživatel je přihlášen a další údaje, které se budou v průběhu aktivit na portálu používat. Tuto funkci má na starost „SESSION“, česky též „relace“, která umožňuje na serveru uchovávat téměř libovolné informace, které se do ní uloží.

Při přihlášení se do session ukládají informace o uživateli, např.:

Jméno, příjmení, uživatelské jméno, práva uživatele, informaci o tom, zda je uživatel člen ústavu, číslo tohoto uživatele apod., viz [Tab. 2].

Tab. 2 - Ukázka uložení session
<pre>\$ _SESSION['prihlasen'] = 1; \$ _SESSION['login'] = \$Result['nick']; \$ _SESSION['UserId'] = \$Result['id']; \$ _SESSION['UserWeb'] = \$Result['web'];</pre>

Tyto údaje nám slouží např. k filtrování stránek, tj. aby uživatel nemohl napsat do příkazové řádky prohlížeče „natvrdo“ cestu k souboru a soubor se mu nezobrazil. Vše je řešeno podmínkou, která ověří, zda příslušný soubor má pro možnost otevření práva nižší než role přihlášeného uživatele, v opačném případě systém informuje uživatele o chybě, která je zobrazena na [Obr. 11], s důvodem, který zamezil přístup na danou stránku.



Nemáte práva pro přístup na tuto stránku

Obr. 11 - Chybová hláška způsobená neoprávněným přístupem

4.1.3 Práce s položkami

Funkce realizující práci s položkami jsou na portále nejdůležitější, neboť je žádoucí, aby majitel položky či *administrátor* byl schopný danou položku vypůjčit, vrátit, změnit její dosavadní údaje či ji vyřadit.

K daným funkcím má přístup přihlášený uživatel s minimálními nutnými právy *člena*. Po přihlášení se v levém menu objeví možnost **ÚPRAVY** (viz [Obr. 12]), která po rozbalení ukrývá všechny již zmiňované funkce a další, které nebyly dosud popsány.



Obr. 12 - Menu umožňující spravovat položky

V horní části tohoto menu se nachází speciální odkaz – znak „-“, který po kliknutí zavolá funkci JQuery a dané menu se „zabalí“. Po dalším stisku této ikonky se menu opět „rozbalí“. Funkce spolupracuje s tzv. „COOKIES“, což jsou uživatelská data, které jsou zaslány do paměti počítače ze strany serveru. Touto metodou je řešeno i zapamatování stavu tohoto menu – zda je sbalené nebo rozbalené, což zajišťuje, že při obnovení stránky menu bude ve stavu, v jakém bylo před obnovou. Uložení základních informací je zobrazeno v [Tab. 3].

Tab. 3 - Práce s cookies

```
$(".column").each(function(index, value) {  
    var colid = value.id;  
    var cookieName = "cookie-" + colid  
    var cookie = $.cookie(cookieName);  
    if ( cookie == null ) { return; }  
    var IDs = cookie.split(",");
```

Základní funkcí je samozřejmě možnost přidat položku, což umožňuje přihlášenému uživateli vložit do databáze záznam, který prezentuje položku s určitým umístěním, daným typem či určitým majitelem. V případě, že přihlášený uživatel má práva *administrátora*, je mu umožněno přidání položky konkrétnímu registrovanému členovi, v opačném případě je má uživatel možnost přidat položku pouze do svého vlastnictví.

V případě, že během určité doby došlo k jistým změnám dané položky – např. byla zjištěna špatně vložená cena, popis, či typ, je na portálu implementována volba, která umožňuje editovat konkrétní položku. V případě uživatelské role *Administrátora* může přihlášený uživatel editovat veškeré dostupné položky, v opačném případě je uživateli povoleno editovat pouze svoje vlastní položky. Výběr položky se provádí pomocí knihovny JQuery, konkrétně našeptávací funkcí, která v závislosti na vložených slovech či písmenech vybírá z databáze zadané položky v reálném čase bez nutnosti aktualizovat stránku či odesílat formulář. Položky se nevybírají v závislosti na pořadí, ale na obsahu písmen či slov, což zaručuje, že v případě zapomenutí prvních písmen dané položky zde zůstává možnost vypsát si informace o dané položce i pomocí určitých písmen, která známe. Našeptávací políčko s výpisem položek obsahující písmenko „W“ je vidět na obrázku [Obr. 13].

Obr. 13 - Našeptávač

Dalšími funkcemi, které spolu blízce souvisí, jsou funkce vypůjčení a vrácení položky. Uživateli je umožněno svoji položku (v případě role *administrátora* všech položek) vypůjčit. Důležité je uchovávat čas, kdy byla položka vypůjčena, komu je vypůjčena, aby bylo dohledatelné, kdo měl danou položku v určitém časovém intervalu či zda je položka již vrácena. Všechny operace, které se provedou s položkou, se zobrazují do speciálního výpisu *Vypůjčených položek*, které je dostupné z menu, viz [Obr. 14].

SUBId	Inv. číslo	Zapůjčil-kdo	Zapůjčeno-komu	Vlastník:	Datum vypůjčky:	Vráceno:	Vrátil-kdo:	Datum vrácení:
760	SW000589	Miroslav Holada	Josef Chaloupka	Zdeněk Plíva	30.03.2012 00:35:05	NE	ZATIM NEVRÁCENO	ZATIM NEVRÁCENO

Obr. 14 - Vypůjčené položky

Doplňující funkcí je i možnost položku vrátit (v případě, že je vypůjčená). V daném políčku se vypisují **pouze** ty položky, které jsou v danou chvíli vypůjčeny, což zajišťuje, že položka, co není vypůjčena, nemůže být vrácena.

Vstupní pole umožňující vypůjčení a vrácení položky naleznete na [Obr. 15] a [Obr. 16].

Obr. 15 - Vypůjčení položky

Položka:

OE00090

Vrátit

Obr. 16 - Vrácení položky

Téměř poslední funkcí, která umožňuje určitým způsobem ovlivnit stav položek, je možnost *vyřazení položky*, kterou uživatel použije v případě, že daná položka nesplňuje daná kritéria používání na pracovišti – jak z hlediska bezpečnosti, tak např. i z hlediska stavu (rozbitá položka). Funkce je opět rozdělena na základě uživatelských rolí – *Administrátor* smí vyřazovat všechny položky, uživatel s nižšími právy smí pouze vyřadit svoje položky.

Poslední funkcí, která zajišťuje operace s položkami, je ryze administrátorská, tzv. *Likvidace položky*, což znamená kompletní *likvidaci* položky z evidence, pro systém daná položka přestane existovat – uvolní se i dané inventární číslo položky pro další možné použití, smaže se ze serveru i fotografie (pokud byla k položce přiřazena).

4.1.4 Strukturované výpisy dat

Veškeré výpisy dat jsou prakticky nepostradatelné, neboť bez nich by uživatel, který aplikaci ovládá, nevěděl o daných položkách nic, kromě těch, které sám editoval či přidával.

Záznamy se zobrazují do přehledně uspořádaných tabulek, kde každý sloupec je důležitý. Některý sloupec popisuje majitele položky, člověka, který provedl editaci či uživatele, který položku vypůjčil. Dalšími sloupci ve strukturovaných výpisech jsou tzv. položkové výpisy, které vypisují informace ohledně dané položky – její cenu, datum změny, umístění, typ apod.

Nejdůležitější funkcí, která využívá tyto výpisy dat, je samotný *log*, který umožňuje vypisovat všechny informace o položkách, které jsou uloženy v databázi. V této funkci je implementována i možnost skrytí daného sloupce, viz [Obr. 17] v závislosti na tom, k čemu data, která se zobrazí, budou sloužit.



[Datum posl.změny](#) | [Majitel](#) | [Typ](#) | [Místnost](#) | [Dar](#) | [Stav](#) | [Posl.editace](#) |

Obr. 17 - Lišta, umožňující skrývání sloupců

Zajímavostí je i strukturované stránkování, které jsem se svolením autora v práci využil.

Jedná se o volně dostupnou třídu, napsanou v jazyce PHP pod GNU / GPL licenci, viz [1], která umožňuje nastavit možnost stránkování (typ apod.).

Funkce přímo pracuje s databází, kde si SQL dotazem zjistí celkový počet záznamů, které chce uživatel vypsat, a celou kolekci dat rozdělí do *n-stránek* viz [Obr. 18].

[<<](#) [<](#) [1](#) [2](#) ... [5](#) [6](#) [7](#) [8](#) [9](#) ... [54](#) [55](#) [>](#) [>>](#)

Obr. 18 - Stránkování s rozdělením po částech

Tabulka, do které se záznamy zobrazují, je stylována pomocí kaskádových stylů, kde úvodní řádek tabulky má odlišnou barvu pozadí, aby vynikl rozdíl mezi záznamy a nadpisy tabulky. Údaje o položkách se vypisují do pevně dané výšky řádku, což zajišťuje stejné rozměry při libovolné položce, viz [Obr. 19].

Inv. číslo	Název	Cena	Datum změny	Místnost	Dar	Stav	Editoval	Edit.
DM500005	nabíječka 9 v quattro	1822	26.09.2011 08:55:00	A01057	NE	Dostupná	Mir. Holada	
DM602820	projektor hitachi ed-x4	22730	26.09.2011 08:55:01	A01057	NE	Dostupná	Mir. Holada	
DM703366	skříňka sn321 se sok	2279	26.09.2011 08:55:02	A01057	NE	Dostupná	Mir. Holada	
DM803390	židle marea lx 1	1684	26.09.2011 08:55:03	A01057	NE	Dostupná	Mir. Holada	

Obr. 19 - Strukturovaný výpis logu

Úvodní řádek tabulky, zejména popisky sloupců, jsou aktivní – tzn. v případě, že uživatel klikne na určitý atribut, upraví se výpis položek v závislosti na tomto atributu, např. v případě, že uživatel chce zobrazit položky dle ceny, stačí kliknout na atribut „cena“ v tabulce. Po dalším kliknutí na tento odkaz se výpis vrátí do původní výchozí pozice – tzn. kliknutím na odkaz, se střídají výpisy záznamů vzestupně a sestupně.

V logu je též možnost editovat položku přímo z tohoto výpisu přes odkaz, který se zobrazuje jako ikonka tužky, v opačném případě, kdy je položka vypůjčena či vyřazena, je možnost editace zamezena a uživateli je zobrazen smutný smajlík.

Další stránkou, která využívá strukturované stránkování, je výpis aktivních položek, tzn. položek, které nejsou vypůjčeny a vyřazeny.



Tato funkce má navíc oproti logu implementován filtr, který umožňuje uživateli vypsat jen položky, o které má zájem – např. položky s cenou od 1 do 1000 Kč, položky určitého typu, s určitým majitelem či položky, které obsahují určité znaky v inventárním čísle.

Funkce jako jediná na portálu zobrazuje danou položku včetně jejího obrázku, který lze kliknutím na ikonku *plus* s textem *Zobrazit obrázky*, viz [Obr. 20].



Obr. 20 - Zobrazení obrázků

Pro zobrazení obrázku, resp. zvětšení obrázku je použita javascriptová funkce, tzn. „Lightbox“, která je vyvíjena pod licencí MIT (viz [7]).

Pro stránkování je zde použita opět volně šiřitelná třída, popsaná výše.

V případě, že uživatel zvolil možnost filtrovat data a zadal určité parametry, systém je vyhodnotí a vypíše je jako informativní výpis, viz [Obr. 21].

Celkem položek: (62)

Cena: 1 do 1000 Kč

Typ: Nábytek

Majitel(é):

Zdeněk Plíva

Obr. 21 - Výpis dle zadaného filtru

Ve spodní části stránky nad samotným výpisem položek se objeví odkaz pro smazání daného filtrování, viz [Obr. 22].



Obr. 22 - Smazat filtrování

Výpis položek je strukturován do tabulky, opět s odlišeným prvním řádkem, kde jsou pojmenovány sloupce. Oproti logu je těchto sloupců méně (inventární číslo, název, informace).

Sloupec „informace“ slouží pro zobrazení detailů o dané položce – výpisu dosáhne uživatel kliknutím na text *Informace* na řádku položky, která ho zajímá. Menu, které se uživateli objeví po animaci, je realizováno knihovnou JQuery UI, efektem **fader**.



Výsledné zobrazení umožňuje uživateli vybrat nezávisle na sobě konkrétní položky, zjistit jejich informace a některé z nich nechat otevřené, či určité zavřít.

V případě, že uživatel nepotřebuje znát detaily o položce, zobrazí se mu klasický výpis, viz [Obr. 23].

Inv. číslo	Název	Informace
DP014531103	zidle kovova 1	Informace
DP014531104	zidle kovova 2	Informace

Obr. 23 - Položky bez zobrazeného detailu

V opačném případě si uživatel vypíše detail dané položky, viz [Obr. 24].

Inv. číslo	Název	Informace
DP014531103	zidle kovova 1	Informace Cena: 978kč Datum editace: 26.09.2011 Posl.uprava:26.09.2011 - 08:55:40 Majitel: Zdeněk Plíva Typ:Nábytek Umístění:A01057 Dar:NE Název:zidle kovova 1 Upravit:Mir.Holada Vypůjčit Vyřadit

Obr. 24 - Detail položky DP014531103

Inventární číslo položky je, stejně jako u funkce *log*, aktivní - po kliknutí se uživatel dostane na detailní výpis všech změn, které jsou v rámci databáze u dané položky dostupné.



4.1.5 Synchronizace

Velmi komplexní funkce, která zajišťuje sjednocení záznamu mezi lokálními daty na tomto portálu a kompletní inventurou (která tyto data může i nemusí obsahovat), která by mohla být dostupná v konkrétních časových intervalech vedením univerzity.

Funkce se stará o aktualizaci dat, která byla oproti lokálním datům změněna, zajišťuje též vyřazení položky a následnou likvidaci v případě, že daná položka je na lokálním portálu dostupná a v kompletní inventuře o této položce nejsou žádné záznamy.

Administrátor vloží do připravené tabulky data a spustí funkci, která tyto metody provede a výsledně oznámí uživateli, které položky byly vyřazeny, přidány a editovány.

4.1.6 Uživatelé

Uživatelé, kteří se pohybují na portálu a aktivně ho využívají, jsou jeho důležitou částí portálu, protože bez uživatelů by nebylo možno tento portál ovládat a pracovat s ním. Práce s rolemi spočívá především v možnosti konkrétního uživatele editovat – změnit mu kompletně údaje jako jméno, příjmení a další data, která si uživatel zadával při registraci. Jediným omezením systému je zamezení editace uživatelského jména a hesla. Přihlášený uživatel (pokud se nejedná o roli *Administrátor*) smí editovat pouze svoje údaje včetně změny hesla.

4.1.7 Generování PDF souborů – mPDF

mPDF [5] je třída, která umožňuje efektivně vytvářet PDF soubory ze zdrojových dat, která jsou k dispozici. Třída mPDF je napsána v jazyce PHP a ke své funkci využívá nejenom podobnou třídu FPDF, ale např. i UFPDF, což ulehčuje vytváření dokumentů s diakritikou s kódováním UTF-8.

Princip generování PDF spočívá ve vytvoření pomyslné webové stránky se všemi náležitostmi – stylování, tabulky, odstavce, barvy...

Jediným rozdílem oproti klasickému výpisu na stránku se výstup přesměruje speciální funkcí jako parametr této třídy, ze které se vytvoří příslušný dokument ze zadaného textu. Nesmírnou výhodou je možnost stylování na stránce, kde tento styl, který byl navržen, PDF dědí.

Portál má implementovanou funkci na generování dostupných položek – všech, které nejsou vypůjčeny, a samozřejmě i inverzní funkci – generují se položky, které byly, či stále jsou vypůjčeny.

4.1.8 Správa fotografií

Každá položka se při vlastním přidání stává součástí systému a jako taková musí mít fotografii. V případě, že daná věc není vyfotografována, je možnost použít výchozí fotku – „Náhled není k dispozici“. V opačném případě nám zpracování tohoto požadavku ulehčí funkce „uploadu“, který se stará o nakopírování obrázku položky, který vybral uživatel ze svého počítače.

Položka se nakopíruje na web s náhodným názvem, aby nedocházelo k chybám, kdy uživatel nahraje obrázek, který má v názvu např. mezeru apod. Systém si pamatuje cestu k fotografii pouze v případě, že daná fotografie je přiřazena k položce – fotografie byla vybrána při přidání nové položky, v opačném případě je k těmto fotografiím přistupováno jako k souborům v konkrétním adresáři na serveru.

Nahrané fotografie jsou zobrazené na stránce pomocí zformátované tabulky tak, aby bylo jasné dané, jak se položka jmenuje, jak vypadá a ikonkou koše je uživateli řečeno, že se může daná fotografie smazat. Kompletní grafickou podobu této funkce znázorňuje obrázek [Obr. 25].



Obr. 25 - Seznam dostupných obrázků

Každý uživatel, kromě role *hosta*, může fotografie nahrávat, a to proto, že může přidávat do systému i vlastní položky, které fotografii mít mohou.

4.2 Grafický design

Design v podstatě nesouvisí s programováním, nicméně je velice důležitý pro správné zobrazení určitých údajů nebo pro lepší orientaci uživatele.

4.2.1 Návrh designu

Grafická podoba webu byla pro mé kódování navržena v programu Adobe Photoshop CS5 Jiřím Podlipným, se kterým jsem konzultoval rozložení menu a barvy, do kterých bude web navržen.

Design splňuje základní požadavky na funkčnost, ergonomii a působí klidným dojmem. Barvy byly převzaty z původního grafického stylu portálu, který se předělával hlavně díky špatné



ergonomii a nepřehlednosti. Tmavší barvy oči uživatele zklidní, proto se na dané stránce lépe orientuje, a spolu s odstíny modré barvy zajišťují zajímavou kombinaci, která na portálu vynikne.

Grafický vzhled portálu včetně hlavní stránky s formulářem pro přihlášení a grafickou podobu, která bude zobrazena uživateli po přihlášení, naleznete v **Příloha A** a **Příloha B**.

4.2.2 Implementace

Grafická část portálu je realizována pomocí kaskádových stylů ve verzi CSS3.

Tato verze je oproti předchozí rozšířena o mnoho funkcí, které ocení např. programátoři, kteří nejsou příliš zkušení v návrzích grafiky. Mezi hlavní výhody této verze patří nové vlastnosti, reprezentující zakulacené rohy, stíny u písma či obrázků nebo volitelnou průhlednost objektu, proto není třeba řešit tyto grafické prvky v daném bitmapovém editoru.

Na portálu byl použit font *TeXGyreHeros* [9], který je používán v závislosti na uvedené licenci, která je k projektu přiložena. Font byl vybrán kvůli podobnosti k fontu *Helvetica*, který se svým designem na portál hodí.

Tab. 4 - Ukázka css

```
#content
{
    text-align: center;
    min-height: 500px;
    margin-bottom: 50px;
    font-size: 100%;
}

#content .container
{
    margin: 0 auto;
    width: 760px;
    text-align: left;
}
```



5 Závěr

Cílem práce bylo vytvořit funkční webový portál, který klade důraz nejen na ergonomii, ale i na funkčnost celku jako takového. Projekt se podařilo vytvořit v plném rozsahu oproti zadání, v některých částech práce bylo zadání práce i rozšířeno.

Bohužel z nedostatku času nebyla možnost dlouhodobějšího testování přístupů do databáze, testování SQL dotazů a celkové odladění chyb, které se projevují nejvíc v praxi. Pevně doufám, že aplikace bude využívána a stane se oblíbeným nástrojem pro ulehčení práce s inventářem i např. pro studenty, kteří v rámci svých prací či projektů využívají databázi položek dostupných na ústavu.

Portál bych chtěl během dalšího studia reálně v praxi otestovat a navrhnout další funkce, které nyní na portálu zaimplementovány nejsou.



6 Použitá literatura a zdroje

- [1] MALÝ, Jakub. Stránkování v PHP a MySQL univerzálně. MALÝ, Jakub. Jakub Malý [online]. 1.0. 3.5.2006 21:26:56 [cit. 2012-03-14]. Dostupné z: <http://blog.jakubmaly.cz/programovani/php/strankovani-v-php-a-mysql-pohodlne-a-univerzalne.aspx>
- [2] CSS3 Info: Box-shadow, one of CSS3's best new features. CSS3 TEAM. CSS3 Info [online]. 3.6.2011 [cit. 2012-03-14]. Dostupné z: <http://www.css3.info/preview/box-shadow/>
- [3] HASSMAN, Martin. Nové značky HTML5. ZDROJAK.ROOT.CZ. Zdroják.cz: Webové standardy [online]. 29.4.2009 [cit. 2012-03-14]. Dostupné z: <http://zdrojak.root.cz/clanky/nove-znacky-html5/>
- [4] Vícevrstvá architektura: Třívrstvá architektura. In: Wikipedie: Otevřená encyklopedie [online]. 27.6.2009, 9.2.2012 [cit. 2012-03-15]. Dostupné z: http://cs.wikipedia.org/wiki/Vícevrstvá_architektura
- [5] MPDF. MPDF: A PHP class to generate PDF files from HTML with Unicode/UTF-8 and CJK support [online]. 23.6.2010, 15.2.2012 [cit. 2012-03-16]. Dostupné z: <http://www.mpdf1.com/mpdf>
- [6] JQuery: UI. JQUERY UI. [online]. [cit. 2012-03-25]. Dostupné z: <http://jqueryui.com/development>
- [7] MIT license: MIT. MIT license [online]. 17.2.2009 [cit. 2012-03-29]. Dostupné z: <http://www.opensource.org/licenses/mit-license.html>
- [8] GPL license: GPL. GPL license [online]. 11.3.2006 [cit. 2012-03-29]. Dostupné z: <http://www.opensource.org/licenses/GPL-2.0>
- [9] License: TeX Gyre Heros. In: TeX Gyre Heros [online]. 30.9.2006 [cit. 2012-04-01]. Dostupné z: <http://www.dailyfreefonts.com/viewfile.php?id=5951&file=GUST%20e-foundry%20License.txt>



Příloha A

Obr. 26 - Grafická podoba přihlašovacího formuláře



Příloha B

Evidence majetku

O projektu

Kontakt

Popis funkcí

LOGOUT

Mirolav Holada

Administrator

Informace

Změnit heslo

LOG UDÁLOSTÍ

MENU

[A] - Přidat ústav

[A] - Přidat pozici

[A] - Přidat typ položky

[A] - Přidat uklad.

VÝPISY - POLOŽKY

Log událostí

Položky

Dostupné

Vyřazené

Vypůjčené

Historie položky

VÝPISY - OBECNÉ

Uživatelé

Výpis s parametry

ÚPRAVY

Log událostí

Položky

Vypůjčené

Historie položky

Vypůjčené

Historie položky

Vypůjčené

Historie položky

Dostupné

Vyřazené

Inv. číslo	Název	Cena	Datum Změny	Osoba	Typ	Místo	Dar	Stav	Editoval	Edit

Evidence majetku naprogramována Zdeňkem Pohlem
v roce 2012 jako bakalářská práce.

Obr. 27 - Grafická podoba celého portálu



Příloha C

Obsah přiloženého DVD

Přiložené DVD obsahuje hlavní adresář „*Bakalářská práce - Zdeněk Pohl*“, ve kterém se nachází zdrojové kódy systému včetně použitých tříd a grafických prvků.

Důležitý je i adresář „Doplňky“, který obsahuje grafické návrhy stránek ve formátu *PSD*, zálohu celé databáze *inventory.sql* a v neposlední řadě je v tomto adresáři též dostupná elektronická verze tohoto dokumentu pod názvem „*Bakalářská práce - Zdeněk Pohl*“ ve formátech:

- a) doc
- b) docx
- c) pdf